**Deterministic and Nondeterministic Functions**

**SQL Server 2012**

[Other Versions](javascript:;)

Description: http://i.technet.microsoft.com/Areas/Epx/Content/Images/ImageSprite.png

* [SQL Server 2008 R2](http://technet.microsoft.com/en-us/library/ms178091(d=printer,v=sql.105).aspx)
* [SQL Server 2008](http://technet.microsoft.com/en-us/library/ms178091(d=printer,v=sql.100).aspx)
* [SQL Server 2005](http://technet.microsoft.com/en-us/library/ms178091(d=printer,v=sql.90).aspx)

Deterministic functions always return the same result any time they are called with a specific set of input values and given the same state of the database. Nondeterministic functions may return different results each time they are called with a specific set of input values even if the database state that they access remains the same.

There are several properties of user-defined functions that determine the ability of the SQL Server Database Engine to index the results of the function, either through indexes on computed columns that call the function, or through indexed views that reference the function. The determinism of a function is one such property. For example, a clustered index cannot be created on a view if the view references any nondeterministic functions. For more information about the properties of functions, including determinism, see [User-Defined Functions](http://technet.microsoft.com/en-us/library/ms191007.aspx).

This topic identifies the determinism of built-in system functions and the effect on the deterministic property of user-defined functions when it contains a call to extended stored procedures.

[Built-in Function Determinism](javascript:void(0))

You cannot influence the determinism of any built-in function. Each built-in function is deterministic or nondeterministic based on how the function is implemented by SQL Server.

All of the aggregate and string built-in functions are deterministic. For a list of these functions, see [Aggregate Functions (Transact-SQL)](http://technet.microsoft.com/en-us/library/ms173454.aspx) and [String Functions (Transact-SQL)](http://technet.microsoft.com/en-us/library/ms181984.aspx).

The following built-in functions from categories of built-in functions other than aggregate and string functions are always deterministic.

|  |  |  |
| --- | --- | --- |
| ABS | DATEDIFF | POWER |
| ACOS | DAY | RADIANS |
| ASIN | DEGREES | ROUND |
| ATAN | EXP | SIGN |
| ATN2 | FLOOR | SIN |
| CEILING | ISNULL | SQUARE |
| COALESCE | ISNUMERIC | SQRT |
| COS | LOG | TAN |
| COT | LOG10 | YEAR |
| DATALENGTH | MONTH |  |
| DATEADD | NULLIF |  |

The following functions are not always deterministic, but can be used in indexed views or indexes on computed columns when they are specified in a deterministic manner.

|  |  |
| --- | --- |
| **Function** | **Comments** |
| CAST | Deterministic unless used with datetime, smalldatetime, or sql\_variant. |
| CONVERT | Deterministic unless one of these conditions exists:   * Source type is sql\_variant. * Target type is sql\_variant and its source type is nondeterministic. * Source or target type is datetime or smalldatetime, the other source or target type is a character string, and a nondeterministic style is specified. To be deterministic, the style parameter must be a constant. Additionally, styles less than or equal to 100 are nondeterministic, except for styles 20 and 21. Styles greater than 100 are deterministic, except for styles 106, 107, 109 and 113. |
| CHECKSUM | Deterministic, with the exception of CHECKSUM(\*). |
| ISDATE | Deterministic only if used with the CONVERT function, the CONVERT style parameter is specified and style is not equal to 0, 100, 9, or 109. |
| RAND | RAND is deterministic only when a seed parameter is specified. |

All the configuration, cursor, metadata, security, and system statistical functions are nondeterministic. For a list of these functions, see [Configuration Functions (Transact-SQL)](http://technet.microsoft.com/en-us/library/ms173823.aspx), [Cursor Functions (Transact-SQL)](http://technet.microsoft.com/en-us/library/ms186285.aspx), [Metadata Functions (Transact-SQL)](http://technet.microsoft.com/en-us/library/ms187812.aspx), [Security Functions (Transact-SQL)](http://technet.microsoft.com/en-us/library/ms186236.aspx), and [System Statistical Functions (Transact-SQL)](http://technet.microsoft.com/en-us/library/ms177520.aspx).

The following built-in functions from other categories are always nondeterministic.

|  |  |
| --- | --- |
| @@CONNECTIONS | @@TOTAL\_WRITE |
| @@CPU\_BUSY | CURRENT\_TIMESTAMP |
| @@DBTS | GETDATE |
| @@IDLE | GETUTCDATE |
| @@IO\_BUSY | GET\_TRANSMISSION\_STATUS |
| @@MAX\_CONNECTIONS | MIN\_ACTIVE\_ROWVERSION |
| @@PACK\_RECEIVED | NEWID |
| @@PACK\_SENT | NEWSEQUENTIALID |
| @@PACKET\_ERRORS | NEXT VALUE FOR |
| @@TIMETICKS | PARSENAME |
| @@TOTAL\_ERRORS | RAND |
| @@TOTAL\_READ | TEXTPTR |

[Calling Extended Stored Procedures from Functions](javascript:void(0))

Functions that call extended stored procedures are nondeterministic, because the extended stored procedures can cause side effects on the database. Side effects are changes to a global state of the database, such as an update to a table, or to an external resource, such as a file or the network; for example, modifying a file or sending an e-mail message. You should not rely on returning a consistent result set when executing an extended stored procedure from a user-defined function. User-defined functions that create side effects on the database are not recommended.

When called from inside a function, the extended stored procedure cannot return result sets to the client. Any Open Data Services API that returns result sets to the client will have a return code of FAIL.

The extended stored procedure can connect back to SQL Server. However, the procedure cannot join the same transaction as the original function that invoked the extended stored procedure.

Similar to invocations from a batch or stored procedure, the extended stored procedure is executed in the context of the Microsoft Windows security account under which SQL Server is running. The owner of the extended stored procedure should consider this when granting permissions to other users to execute the procedure.